IN THE CLAIMS:

This listing of claims replaces all prior versions of the claims in the application. Please add new claims 55-62. Please cancel claims 4, 12, 19-20, 25, 33, 40, 42 and 52 without prejudice to, or disclaimer of, the subject matter therein. The Applicant reserves the right to pursue the subject matter in these claims in a continuation application. Claims 1-3, 5-11, 13-18, 21-24, 26-32, 34-39, 41, 43-51, and 53-62 are pending in this application with claims 1, 9, 15, 17, 21, 29, 36, 38, 41, 50, and 55-62 being the independent claims. Currently amended claims are shown with additions underlined and deletions in ~~strikethrough~~ text. No new matter is added by these amendments.

1. (Currently Amended) A method, comprising:

receiving data transmitted over a network, the receiving occurring via a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determining whether a Tabular Data Stream (TDS) handshake protocol is required to handle the received data;

initiating a TDS handshake protocol by the client, if it is determined that a TDS handshake protocol is required;

determining whether the at least one portion of the data is encrypted; and

initiating a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

2. (Original) The method of claim 1, wherein the initiating a secure protocol includes initiating a secure socket layer (SSL) protocol.

3. (Currently Amended) The method of claim 1, wherein the client is~~using a standard secure protocol library includes~~ a pure Java client.

4. (Canceled)

5. (Original) The method of claim 1, wherein the standard secure protocol library includes a Java Secure Socket Extension (JSSE).

6. (Original) The method of claim 1, wherein the secure protocol includes a secure socket layer (SSL) protocol, and the standard secure protocol library includes a standard SSL library.

7. (Original) The method of claim 1, wherein the received data is received from a server that uses a structured query language and a TDS handshake protocol.

8. (Original) The method of claim 1, wherein the data is received from a Microsoft Structured Query Language (MS SQL) server.

9. (Currently Amended) A method, comprising:

receiving data transmitted by a Microsoft Structured Query Language (MS SQL) server over a network, the receiving occurring via a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determining whether an MS SQL handshake protocol is required to handle the received data;

initiating an MS SQL handshake protocol by the client, if it is determined that an MS SQL handshake protocol is required;

determining whether the at least one portion of the data is encrypted; and

initiating a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

10. (Original) The method of claim 9, wherein the initiating a secure protocol includes initiating a secure socket layer (SSL) protocol.

11. (Currently Amended) The method of claim 9, wherein the client is~~using a standard secure protocol library includes~~ a pure Java client.

12. (Canceled)

13. (Original) The method of claim 9, wherein the standard secure protocol library includes a Java Secure Socket Extension (JSSE).

14. (Original) The method of claim 9, wherein the secure protocol includes a secure socket layer (SSL) protocol, and the standard secure protocol library includes a standard SSL library.

15. (Currently Amended) A processor-readable medium storing code representing instructions to cause a processor to perform a process, the code comprising code to:

receive data transmitted over a network via a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determine whether a Tabular Data Stream (TDS) handshake protocol is required to handle the received data;

initiate a TDS handshake protocol by the client, if it is determined that a TDS handshake protocol is required;

determine whether the at least one portion of the data is encrypted; and

initiate a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

16. (Original) The processor-readable medium of claim 15, wherein the client using a standard secure protocol library includes a pure Java client using a standard secure socket layer (SSL) library.

17. (Currently Amended) A processor-readable medium storing code representing instructions to cause a processor to perform a process, the code comprising code to:

receive data transmitted by a Microsoft Structured Query Language (MS SQL) server over a network via a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determine whether an MS SQL handshake protocol is required to handle the received data;

initiate an MS SQL handshake protocol by the client, if it is determined that an MS SQL handshake protocol is required;

determine whether the at least one portion of the data is encrypted; and

initiate a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

18. (Original) The processor-readable medium of claim 17, wherein the client using a standard secure protocol library includes a pure Java client using a standard secure socket layer (SSL) library.

19.-20. (Canceled)

21. (Currently Amended) A method, comprising:

preparing data to be transmitted to a server from a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determining if a Tabular Data Stream (TDS) handshake protocol is required to communicate with the server;

initiating a TDS handshake protocol, if it is determined that a TDS handshake protocol is required;

determining if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiating the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

22. (Original) The method of claim 21, further comprising:

transmitting encrypted data using the standard secure protocol.

23. (Original) The method of claim 21, wherein the initiating the standard secure protocol includes initiating a secure socket layer (SSL) protocol.

24. (Currently Amended) The method of claim 21, wherein the client is~~using a standard secure protocol library includes~~ a pure Java client.

25. (Canceled)

26. (Original) The method of claim 21, wherein the standard secure protocol library includes a Java Secure Socket Extension (JSSE).

27. (Original) The method of claim 21, wherein the standard secure protocol includes a secure socket layer (SSL) protocol, and the standard secure protocol library includes a standard SSL library.

28. (Original) The method of claim 21, wherein the data is received from a Microsoft Structured Query Language (MS SQL) server.

29. (Currently Amended) A method, comprising:

preparing data to be transmitted to a server from a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determining if a Microsoft Structured Query Language (MS SQL) server handshake protocol is required to communicate with the server;

initiating an MS SQL handshake protocol, if it is determined that an MS SQL handshake protocol is required;

determining if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiating the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

30. (Original) The method of claim 29, further comprising:

transmitting encrypted data using the standard secure protocol.

31. (Original) The method of claim 29, wherein the initiating the standard secure protocol includes initiating a secure socket layer (SSL) protocol.

32. (Currently Amended) The method of claim 29, wherein the client is~~using a standard secure protocol library includes~~ a pure Java client.

33. (Canceled)

34. (Original) The method of claim 29, wherein the standard secure protocol library includes a Java Secure Socket Extension (JSSE).

35. (Original) The method of claim 29, wherein the standard secure protocol includes a secure socket layer (SSL) protocol, and the standard secure protocol library includes a standard SSL library.

36. (Currently Amended) A processor-readable medium storing code representing instructions to cause a processor to perform a process, the code comprising code to:

prepare data to be transmitted to a server from a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determine if a Tabular Data Stream (TDS) handshake protocol is required to communicate with the server;

initiate a TDS handshake protocol, if it is determined that a TDS handshake protocol is required;

determine if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiate the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

37. (Original) The processor-readable medium of claim 36, wherein the client using a standard secure protocol library includes a pure Java client using a standard secure socket layer (SSL) library.

38. (Currently Amended) A processor-readable medium storing code representing instructions to cause a processor to perform a process, the code comprising code to:

prepare data to be transmitted to a server from a Java client using a standard secure protocol library including a Java Database Connectivity (JDBC) Type 4 driver;

determine if a Microsoft Structured Query Language (MS SQL) server handshake protocol is required to communicate with the server;

initiate an MS SQL handshake protocol, if it is determined that an MS SQL handshake protocol is required;

determine if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiate the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

39. (Original) The processor-readable medium of claim 38, wherein the client using a standard secure protocol library includes a pure Java client using a standard secure socket layer (SSL) library.

40. (Canceled)

41. (Currently Amended) A system, comprising:

a first client application configured to transmit and to receive secure communications via a network using a standard secure protocol library, the secure communications including queries sent by the first client application and responses received by the first client application;

a server application configured to receive the queries sent by the first client application via the network and to transmit the responses received by the first client application via the network, the server application requiring a proprietary server handshake protocol to communicate with the first client application using a standard secure protocol associated with the standard secure protocol library;

a translation component configured to receive the queries sent by the first client application and to translate the queries into queries that use the proprietary server handshake

protocol of the server application so that they are understandable to the server application, the translation component further configured to receive the responses transmitted by the server application and to translate the responses into responses that do not require use of the proprietary server handshake protocol such that they are understandable by the first client.

42. (Canceled)

43. (Original) The system of claim 41, wherein the standard secure protocol library includes a standard secure socket layer (SSL) library.

44. (Original) The system of claim 41, wherein the first client application includes a pure Java client.

45. (Original) The system of claim 41, wherein the first client application includes a Java client using a Java Database Connectivity (JDBC) Type 4 driver.

46. (Original) The system of claim 41, wherein the standard secure protocol library includes a Java Secure Socket Extension (JSSE).

47. (Original) The system of claim 41, wherein the server application uses a structured query language and the proprietary server handshake protocol includes a Tabular Data Stream (TDS) handshake protocol.

48. (Original) The system of claim 41, wherein the server application is a Microsoft Structured Query Language (MS SQL) server and the proprietary server handshake protocol includes an MS SQL server handshake protocol.

49. (Original) The system of claim 41, wherein the first client application is further configured to communicate using sockets, the server application is configured to communicate using named pipes, and the translation component is configured to translate communications between sockets and named pipes.

50. (Currently Amended) An apparatus, comprising:

a secure data communication object configured to receive secure communication data as input and to output secure communication data;

a socket communication object configured to receive socket communication data as input and to output socket communication data;

a named pipe communication object configured to receive named pipe communication data as input and to output named pipe communication data; and

a translation communication object in communication with each of the secure data communication object, the socket communication object, and the named pipe communication object, the translation communication object configured to translate named pipe communication data received via the named pipe communication object for handling by the secure data communication object, the translation communication object being further configured to send socket communication data received via the socket communication object to the secure data communication object, the translation communication object including a Tabular Data Stream (TDS) handshake object configured to perform a TDS handshake with a TDS client, the TDS handshake object further configured to translate TDS data to a non-TDS data format and non-TDS data to TDS data format.

51. (Original) The apparatus of claim 50, wherein the translation component is further configured to translate secure communication data received via the secure communication object for handling by the named pipe communication object, the translation communication object being further configured to send secure communication data received via the secure communication object to the socket data communication object.

52. (Canceled)

53. (Original) The apparatus of claim 50, wherein the secure communication object includes a Secure Socket Layer (SSL) communication object.

54. (Original) The apparatus of claim 50, wherein the secure communication object includes a Java client using a standard Secure Socket Layer (SSL) library.

55. (New) A method, comprising:

receiving data transmitted over a network, the receiving occurring via a client using a standard secure protocol library, the standard secure protocol library including a Java Secure Socket Extension (JSSE);

determining whether a Tabular Data Stream (TDS) handshake protocol is required to handle the received data;

initiating a TDS handshake protocol by the client, if it is determined that a TDS handshake protocol is required;

determining whether the at least one portion of the data is encrypted; and

initiating a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

56. (New) A method, comprising:

receiving data transmitted by a Microsoft Structured Query Language (MS SQL) server over a network, the receiving occurring via a client using a standard secure protocol library, the standard secure protocol library including a Java Secure Socket Extension (JSSE);

determining whether an MS SQL handshake protocol is required to handle the received data;

initiating an MS SQL handshake protocol by the client, if it is determined that an MS SQL handshake protocol is required;

determining whether the at least one portion of the data is encrypted; and

initiating a secure protocol to handle the at least one portion of the data, if it is determined that the at least one portion of the data is encrypted.

57. (New) A method, comprising:

preparing data to be transmitted to a server from a client using a standard secure protocol library, the standard secure protocol library including a Java Secure Socket Extension (JSSE);

determining if a Tabular Data Stream (TDS) handshake protocol is required to communicate with the server;

initiating a TDS handshake protocol, if it is determined that a TDS handshake protocol is required;

determining if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiating the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

58. (New) A method, comprising:

preparing data to be transmitted to a server from a client using a standard secure protocol library, the standard secure protocol library including a Java Secure Socket Extension (JSSE);

determining if a Microsoft Structured Query Language (MS SQL) server handshake protocol is required to communicate with the server;

initiating an MS SQL handshake protocol, if it is determined that an MS SQL handshake protocol is required;

determining if data to be transmitted is to be encrypted using a standard secure protocol associated with the standard secure protocol library; and

initiating the standard secure protocol, if it is determined that the data is to be encrypted using the standard secure protocol.

59. (New) A system, comprising:

a first client application configured to transmit and to receive secure communications via a network using a standard secure protocol library, the first client application including a Java client using a Java Database Connectivity (JDBC) Type 4 driver, the secure communications including queries sent by the first client application and responses received by the first client application;

a server application configured to receive the queries sent by the first client application via the network and to transmit the responses received by the first client application via the network, the server application requiring a proprietary server handshake protocol to

communicate with the first client application using a standard secure protocol associated with the standard secure protocol library;

a translation component configured to receive the queries sent by the first client application and to translate the queries into queries that use the proprietary server handshake protocol of the server application so that they are understandable to the server application.

60. (New) A system, comprising:

a first client application configured to transmit and to receive secure communications via a network using a standard secure protocol library, the standard secure protocol library including a Java Secure Socket Extension (JSSE), the secure communications including queries sent by the first client application and responses received by the first client application;

a server application configured to receive the queries sent by the first client application via the network and to transmit the responses received by the first client application via the network, the server application requiring a proprietary server handshake protocol to communicate with the first client application using a standard secure protocol associated with the standard secure protocol library;

a translation component configured to receive the queries sent by the first client application and to translate the queries into queries that use the proprietary server handshake protocol of the server application so that they are understandable to the server application.

61. (New) A system, comprising:

a first client application configured to transmit and to receive secure communications via a network using a standard secure protocol library, the secure communications including queries sent by the first client application and responses received by the first client application;

a server application configured to receive the queries sent by the first client application via the network and to transmit the responses received by the first client application via the network, the server application requiring a proprietary server handshake protocol to communicate with the first client application using a standard secure protocol associated with the standard secure protocol library;

a translation component configured to receive the queries sent by the first client application and to translate the queries into queries that use the proprietary server handshake protocol of the server application so that they are understandable to the server application,

the first client application configured to communicate using sockets, the server application is configured to communicate using named pipes, and the translation component is configured to translate communications between sockets and named pipes.

62. (New) An apparatus, comprising:

a secure data communication object configured to receive secure communication data as input and to output secure communication data;

a socket communication object configured to receive socket communication data as input and to output socket communication data;

a named pipe communication object configured to receive named pipe communication data as input and to output named pipe communication data; and

a translation communication object in communication with each of the secure data communication object, the socket communication object, and the named pipe communication object, the translation communication object configured to translate named pipe communication data received via the named pipe communication object for handling by the secure data communication object, the translation communication object being further configured to send socket communication data received via the socket communication object to the secure data communication object, the translation component being further configured to translate secure communication data received via the secure communication object for handling by the named pipe communication object, the translation communication object being further configured to send secure communication data received via the secure communication object to the socket data communication object.